Compiled knowledge of what I need for this project

# Most of the knowledge is now in the knowledge ipynb file

# Contents:

1. [Plotting a gradient](#_1._Plotting_a)

# 1. Plotting a gradient

**1. Basic Model: Exponential Decay with Distance**

A simple model assumes the concentration CCC decreases **exponentially** with the distance rrr from the source:

* C0C\_0C0​: concentration at the source (maximum)
* rrr: distance from the source point
* kkk: decay rate (higher means faster decay)

This is a **radial symmetric** field — the same in every direction.

**2. Computing the Distance**

Let the source be at point (x0,y0)(x\_0, y\_0)(x0​,y0​). The distance from any point (x,y)(x, y)(x,y) is:

r=(x−x0)2+(y−y0)2r = \sqrt{(x - x\_0)^2 + (y - y\_0)^2}r=(x−x0​)2+(y−y0​)2​

# Gausian random field

**Gaussian random field** (GRF) is a [random field](https://en.wikipedia.org/wiki/Random_field) involving [Gaussian probability density functions](https://en.wikipedia.org/wiki/Multivariate_normal_distribution) of the variables.

My code:

# 1. Slit-like repellent field

repellent\_positions = [

    (np.random.uniform(0.0, 0.45) if np.random.rand() < 0.5 else np.random.uniform(0.55, 1.0),

     np.random.uniform(0.0, 0.2))

    for \_ in range(num\_repellents)

]

repellent\_field = np.zeros\_like(x)

for rx, ry in repellent\_positions:

    repellent\_field += repellent\_strength \* np.exp(-((x - rx)\*\*2 + (y - ry)\*\*2) / (2 \* sigma\_repellent\*\*2))